日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

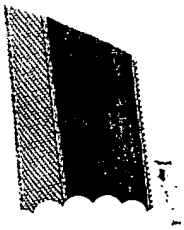
This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日
Date of Application: ２００３年　３月３１日

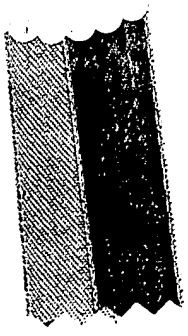
出願番号
Application Number: 特願２００３－０９６０８８

[ST. 10/C]: [ＪＰ２００３－０９６０８８]

出願人
Applicant(s): 株式会社エヌ・ティ・ティ・ドコモ



CERTIFIED COPY OF PRIORITY DOCUMENT



２００４年　３月２６日

特許庁長官
Commissioner,
Japan Patent Office

今井康夫

出証番号　出証特２００４－３０２５４９３

【書類名】　特許願
【整理番号】　DCMH140848
【提出日】　平成15年 3月31日
【あて先】　特許庁長官　殿
【国際特許分類】　H04L 9/00
【発明の名称】　端末装置及びプログラム
【請求項の数】　7
【発明者】
【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
【氏名】　成瀬　直樹
【発明者】
【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
【氏名】　市川　裕一
【発明者】
【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
【氏名】　大井　達郎
【発明者】
【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
【氏名】　渡邉　信之
【発明者】
【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
【氏名】　服部　易憲

【発明者】

【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内

【氏名】　竹下　理人

【発明者】

【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内

【氏名】　西田　真和

【発明者】

【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内

【氏名】　浅井　真生

【発明者】

【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内

【氏名】　津田　雅之

【発明者】

【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内

【氏名】　富岡　淳樹

【発明者】

【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内

【氏名】　山田　和宏

【発明者】

【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内

【氏名】　神谷　大

【発明者】
　【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
　【氏名】　鷲尾　諭
【発明者】
　【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
　【氏名】　山根　直樹
【発明者】
　【住所又は居所】　東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
　【氏名】　村上　圭一
【特許出願人】
　【識別番号】　392026693
　【氏名又は名称】　株式会社エヌ・ティ・ティ・ドコモ
【代理人】
　【識別番号】　100098084
　【弁理士】
　【氏名又は名称】　川▲崎▼　研二
【選任した代理人】
　【識別番号】　100111763
　【弁理士】
　【氏名又は名称】　松本　隆
【手数料の表示】
　【予納台帳番号】　038265
　【納付金額】　21,000円
【提出物件の目録】
　【物件名】　明細書　1
　【物件名】　図面　1

【物件名】　要約書　1

【プルーフの要否】　要

【書類名】　明細書

【発明の名称】　端末装置及びプログラム

【特許請求の範囲】

【請求項１】　他のファイルの予め定められた個所に格納されたデータに基づいて算出された値を表す特定のデータを格納した第１のファイルと、データを格納した第２のファイルを受信する受信手段と、

前記第２のファイルの前記個所に格納されたデータを予め定められた１方向関数に代入して値を算出する算出手段と、

前記算出手段により算出された値と前記特定のデータで表される値とを比較する比較手段と、

前記比較手段による比較結果に基づいて前記第１のファイルと前記第２のファイルとの組み合わせの正当性を判定する判定手段と

を有する端末装置。

【請求項２】　前記個所はファイル全体である

ことを特徴とする請求項１に記載の端末装置。

【請求項３】　前記個所はファイルの作成者を示すデータが格納される個所である

ことを特徴とする請求項１に記載の端末装置。

【請求項４】　前記第２のファイルは前記端末装置により実行されるプログラムを格納したファイルであり、前記第１のファイルは前記第２のファイルをダウンロードするために必要なデータを格納したファイルである

ことを特徴とする請求項１に記載の端末装置。

【請求項５】　前記第２のファイルは前記端末装置により実行されるプログラムを格納したファイルをダウンロードするために必要なデータを格納したファイルであり、前記第１のファイルは前記端末装置が前記プログラムを実行することにより実現されるアプリケーションの機能の制限に必要なデータを格納したファイルである

ことを特徴とする請求項１に記載の端末装置。

【請求項６】　前記１方向関数はハッシュ関数である

ことを特徴とする請求項１に記載の端末装置。

【請求項７】　コンピュータ装置を、

他のファイルの予め定められた個所に格納されたデータに基づいて算出された値を表す特定のデータを格納した第１のファイルと、データを格納した第２のファイルを受信する受信手段と、

前記第２のファイルの前記個所に格納されたデータを予め定められた１方向関数に代入して値を算出する算出手段と、

前記算出手段により算出された値と前記特定のデータで表される値とを比較する比較手段と、

前記比較手段による比較結果に基づいて前記第１のファイルと前記第２のファイルとの組み合わせの正当性を判定する判定手段

として機能させるためのプログラム。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、ダウンロードしたデータの正当性を判定する技術に関する。

【０００２】

【従来の技術】

近年、プログラム等のソフトウェアをインターネット等のネットワーク経由で通信端末にダウンロードして使用することが広く行われている。このような環境下でソフトウェアの改竄や「なりすまし」などの不正行為を完全に排除することは困難であるから、不正なソフトウェアが通信端末にダウンロードされてしまう虞がある。このような事情から、ダウンロードしたソフトウェアの正当性を確認するための技術が提案されている。例えば、ダウンロードするソフトウェアのハッシュ値を記録したＩＣ（Integrated Circuit）カードを当該ソフトウェアの提供元が予めユーザに貸与する方法が提案されている（特許文献１参照）。この方法では、ユーザが通信端末にＩＣカードを装填しソフトウェアのダウンロードを指示すると、通信端末はソフトウェアをダウンロードし、ハッシュ関数を用いて当該ソフトウェアに対するハッシュ値を算出し、このハッシュ値をＩＣカードに

記録されたハッシュ値と比較し、両者が一致したら、受信したソフトウェアが正当であると判断する。

【０００３】

【特許文献】

特開平１１－２０５７６７号公報

【０００４】

【発明が解決しようとする課題】

ところで、Ｊａｖａ－ＡＰ（アプリケーション）ソフトウェアをダウンロードし実行することができる携帯電話機が普及している。この種の携帯電話機へＪａｖａ－ＡＰソフトウェアをダウンロードする際には、WWW（World Wide Web）を構成するサーバ装置からＡＤＦ（Application Descrcptor File）がダウンロードされ、次いでＪＡＲ（Java Archive）ファイルがダウンロードされることになる。これらのファイルも不正行為の対象となり得るから、ダウンロードしたソフトウェアの正当性の確認が必要である。

ところで、ＡＤＦは対応するＪＡＲファイルの作成日付などのＪＡＲファイルに関する情報を内包していることから、対応するＪＡＲファイルが更新されると更新されねばならない。つまり、ＡＤＦとＪＡＲファイルには正当な組み合わせが存在する。したがって、Ｊａｖａ－ＡＰソフトウェアの正当性の確認には、ＡＤＦ及びＪＡＲファイルの組み合わせの正当性の確認が必須である。

ＡＤＦ及びＪＡＲファイルの組み合わせの正当性を確認する方法としては、正当な組み合わせのＡＤＦおよびＪＡＲファイルを一体化してからハッシュ値を算出し、このようなハッシュ値を特許文献に記載された技術において用いるようにする方法が考えられる。しかし、Ｊａｖａ－ＡＰソフトウェアは、その提供者によりバグが修正されたりバージョンアップされたりするものであるから、その度にハッシュ値が変わることになる。したがって、Ｊａｖａ－ＡＰソフトウェアの提供者であるＣＰ（Contents Provider）は、当該Ｊａｖａ－ＡＰソフトウェアの変更の都度、ハッシュ値を記録したＩＣカードを当該Ｊａｖａ－ＡＰソフトウェアの利用者へ配布しなければならない。これは非現実的である。

【０００５】

本発明は、上述した事情に鑑みて為されたものであり、ダウンロードした関連する複数のファイルの組み合わせの正当性を容易に判定することができる技術を提供することを目的としている。

【０００６】

【課題を解決するための手段】

上述した課題を解決するために、本発明は、他のファイルの予め定められた個所に格納されたデータに基づいて算出された値を表す特定のデータを格納した第１のファイルと、データを格納した第２のファイルを受信する受信手段と、前記第２のファイルの前記個所に格納されたデータを予め定められた１方向関数に代入して値を算出する算出手段と、前記算出手段により算出された値と前記特定のデータで表される値とを比較する比較手段と、前記比較手段による比較結果に基づいて前記第１のファイルと前記第２のファイルとの組み合わせの正当性を判定する判定手段とを有する端末装置を提供する。

また、本発明は、コンピュータ装置を、他のファイルの予め定められた個所に格納されたデータに基づいて算出された値を表す特定のデータを格納した第１のファイルと、データを格納した第２のファイルを受信する受信手段と、前記第２のファイルの前記個所に格納されたデータを予め定められた１方向関数に代入して値を算出する算出手段と、前記算出手段により算出された値と前記特定のデータで表される値とを比較する比較手段と、前記比較手段による比較結果に基づいて前記第１のファイルと前記第２のファイルとの組み合わせの正当性を判定する判定手段として機能させるためのプログラムを提供する。

【０００７】

本発明によれば、受信手段により受信された第２のファイルの予め定められた個所に格納されたデータを１方向関数に代入して算出された値と、受信手段により受信された第１のファイルに格納された特定のデータで表される値とが比較され、この比較結果に基づいて前記第１のファイルと前記第２のファイルとの組み合わせの正当性が判定される。

【０００８】

【発明の実施の形態】

以下、図面を参照して、本発明の実施の一形態である配信システムについて図面を参照して説明する。なお、図面において、共通する部分には同一の符号が付されている。

この配信システムは、ユーザが、携帯電話機を操作して所望のＪａｖａ－ＡＰソフトウェアを携帯電話機にダウンロードおよびインストールし、携帯電話機において起動するためのものである。

【０００９】

本システムにおけるＪａｖａ－ＡＰソフトウェアのダウンロードは、まず、携帯電話機が、Ｊａｖａ－ＡＰソフトウェアの内容を説明した画面を表示した後、この携帯電話機のユーザが所望するＪａｖａ－ＡＰソフトウェアに対応したＡＤＦを受信し、次いで、上記Ｊａｖａ－ＡＰソフトウェアに対応したＳＤＦ（セキュリティ記述ファイル）と称せられるファイルを受信し、最後にＪＡＲファイルを受信するという手順で行われる。ここで、ＳＤＦは、携帯電話機内におけるＪａｖａ－ＡＰソフトウェアの挙動を制限する内容が記述されたファイルである。よって、携帯電話機は、インストールしたＪａｖａ－ＡＰソフトウェアを実行するに際しては、このＳＤＦの記述内容に従うこととなる。このＳＤＦは、Ｊａｖａ－ＡＰソフトウェアについて上記通信事業者とこのＪａｖａ－ＡＰソフトウェアを提供するＣＰとの間で結ばれた契約に従って通信事業者により作成される。

ところで、本実施形態では、対応するＳＤＦが存在するＪａｖａ－ＡＰソフトウェアと、対応するＳＤＦが存在しないＪａｖａ－ＡＰソフトウェアとが用意されている。前者のＪａｖａ－ＡＰソフトウェアは、対応するＳＤＦに記述された許可情報による挙動制限を受けるものであり、通信事業者がＣＰとの契約に基づいて信頼性を保証したものであることから、以降の説明では、前者を「トラステッドＪａｖａ－ＡＰソフトウェア」と呼ぶ。これに対応して、後者を「非トラステッドＪａｖａ－ＡＰソフトウェア」と呼ぶ。

なお、本実施形態の説明において「Ｊａｖａ－ＡＰソフトウェア」と言う場合には、それが「トラステッドＪａｖａ－ＡＰソフトウェア」であればＡＤＦ、ＳＤＦ及びＪＡＲファイルを含む概念とし、「非トラステッドＪａｖａ－ＡＰソフトウェア」であればＡＤＦ及びＪＡＲファイルを含む概念とする。

【００１０】

（１：構成）

図１に示されるように、この配信システムは、インターネット１１に接続されたＣＰサーバ装置１２と、通信事業者が移動パケット通信サービスを提供するために用いる移動パケット通信網１５と、この移動パケット通信網１５を介して通信相手とパケット通信を行う携帯電話機１６と、インターネット１１と移動パケット通信網１５とを相互接続するゲートウェイサーバ装置１７と、ゲートウェイサーバ装置１７に接続されたトラステッドサーバ装置１８とを有する。この配信システムには多数の携帯電話機が存在するが、図面が繁雑になるのを避けるために一つの携帯電話機１６のみが図示されている。これと同様の理由により、１つのＣＰサーバ装置１２のみが図示されている。

【００１１】

以下、この配信システムの各構成要素について詳細に説明する。

（１－１：ＣＰサーバ装置）

ＣＰサーバ装置１２は、一般的なWWWサーバ装置と同様のハードウェアおよび機能を有する。また、ＣＰサーバ装置１２はハードディスク装置１２Ａを有する。ＣＰサーバ装置１２は、ＴＣＰ（Transmission Control Protocol）に従ったコネクション（以後、ＴＣＰコネクション）を通信相手との間に確立し、このコネクションを介してＨＴＴＰ（Hypertext Transfer Protocol）のＧＥＴメソッドを用いた要求メッセージを受信すると、このＧＥＴメソッドに指定されたＵＲＬ（Uniform Resource Locator）で特定されるファイルを自身のハードディスク装置１２Ａから読み出し、このファイルを含むＨＴＴＰの応答メッセージを返送してこのコネクションを切断する。

【００１２】

ハードディスク装置１２Ａは、Ｊａｖａプログラミング言語を用いて作成されたプログラムを内包するＪＡＲファイルと、このＪＡＲファイルに関する情報を記述したＡＤＦを記憶し得る。

ＣＰサーバ装置１２に記憶され得るＡＤＦには、トラステッドＪａｖａ－ＡＰソフトウェアに対応したＡＤＦと、非トラステッドＪａｖａ－ＡＰソフトウェア

に対応したＡＤＦとがある。これらのいずれのＡＤＦにおいても、Ｊａｖａ－ＡＰソフトウェアの名称や、WWWにおけるＪＡＲファイルの記憶位置を示すＪＡＲ保存先ＵＲＬデータや、ＪＡＲファイルのサイズを示す情報や、ＪＡＲファイルの最終変更日時を示す情報等の、従来からＡＤＦに内包されている情報が記述されている。これに加えて、トラステッドＪａｖａ－ＡＰソフトウェアに対応したＡＤＦは、図２に示されるように、トラステッドＪａｖａ－ＡＰソフトウェアである場合に記述されるトラステッドＡＰＩＤデータ、ＳＤＦがWWWにおいて記憶されている位置を示すトラステッドサーバドメインと、図示せぬＣＡ（認証局；Certificate Authority）からＣＰサーバ装置１２を運用するＣＰへ提供された証明書データと、ＪＡＲファイルのハッシュ値を表すＪａｒハッシュ値データが内包されている。

ここで、ハッシュ値とは、任意のデータをハッシュ関数に代入することにより算出される一定長の算出値である。ハッシュ関数は一方向関数の一種である。「一方向関数」とは、$y = f(x)$ は高速に計算できるが，y から x を求める f の逆関数は存在せず， x を求めるには膨大な計算時間を要し、xを求めるのは事実上不可能な関数をいう。

また、ＣＰサーバ装置１２は、ＣＰの指示に従って上記各ファイルを作成および更新する機能を備えている。また、ハードディスク装置１２Ａは、ＣＰが認証局に認証されていることを証明するための、ＣＡが発行した証明書データを記憶し得る。また、ＣＰサーバ装置１２は、ＪＡＲファイルや証明書データから、ＳＨＡ－１のハッシュアルゴリズムに従ってハッシュ値を算出するプログラムを記憶し得る。

【００１３】

（１－２：ゲートウェイサーバ装置）

ゲートウェイサーバ装置１７は、前述の通信事業者により管理されており、移動パケット通信網１５とインターネット１１とを接続する一般的なゲートウェイサーバ装置と同様の構成を有し、移動パケット通信網１５とインターネット１１との間で相互に通信を中継する。

【００１４】

（１－３：トラステッドサーバ装置）

トラステッドサーバ装置１８は前述の通信事業者により管理されており、ＷＷＷを構成し、一般的なＷＷＷサーバ装置と同様のハードウェアおよび機能を有する。また、トラステッドサーバ装置１８はハードディスク装置１８Ａを有し、ＴＣＰコネクションを通信相手との間に確立し、このコネクションを介してＨＴＴＰのＧＥＴメソッドを用いた要求メッセージを受信すると、このＧＥＴメソッドに指定されたＵＲＬで特定されるファイルをハードディスク装置１８Ａから読み出し、このファイルを含むＨＴＴＰの応答メッセージを返送してこのコネクションを切断する。

【００１５】

ハードディスク装置１８Ａに記憶されるファイルとしては、複数のトラステッドＪａｖａ－ＡＰソフトウェアに対応した複数のＳＤＦがある。

ＳＤＦは、トラステッドＪａｖａ－ＡＰソフトウェア毎に通信事業者により作成されるファイルである。図３に示されるように、ＳＤＦは、トラステッドＪａｖａ－ＡＰソフトウェアが記憶されている位置を表すＪＡＲ保存先ＵＲＬデータと、対応するＡＤＦファイルに証明書データが内包されているか否かを表すＡＤＦ証明書有無フラグデータと、ＡＤＦに含まれている証明書データから算出したハッシュ値を表す証明書ハッシュ値データと、トラステッドＪａｖａ－ＡＰソフトウェアが使用を許可されているＡＰＩ（Application Program Interface）やＵＲＬを表すパーミッション情報データとを内包している。

ここで、パーミッション情報データには、電話帳参照、未読メール取得、発着信情報取得ＡＰＩの使用を許可するか否かを表す個人情報取得データや、着信メロディ、発着信画像、待ち受け画像登録など、携帯電話機設定更新用ＡＰＩの使用を許可するか否かを表す設定更新データや、アクセスを許可するＵＲＬを表すアクセス許可ＵＲＬデータが含まれている。

【００１６】

（１－４：携帯電話機）

携帯電話機１６は、図４に示されるように、ＯＳ（オペレーティングシステム）ソフトウェア、Ｊａｖａ－ＡＰを実行する環境を構築するためのＪａｖａ－Ａ

Ｐ環境ソフトウェア、および各種ネイティブＡＰソフトウェア等を記憶したＲＯＭ１６Ａと、ＲＯＭ１６Ａからプログラムを読み出して実行するＣＰＵ１６Ｂと、表示部１６Ｃと、不揮発性メモリ１６Ｄと、ＲＡＭ１６Ｅと、通信部１６Ｆと、操作部１６Ｇとを有し、これらはバスによって接続されている。

【００１７】

表示部１６Ｃは、例えば液晶表示パネルやパネル駆動回路を有し、ＣＰＵ１６Ｂから供給されるデータで表される画像を表示する。

不揮発性メモリ１６Ｄは、例えば、ＳＲＡＭ（Static Random Access Memory）やＥＥＰＲＯＭ（Ellectrically Erasable and Programmable Read Only Memory）である。また、この不揮発性メモリ１６Ｄは、WWWを構成するサーバ装置からダウンロードされたＪａｖａ－ＡＰソフトウェアを記憶するために使用される。

また、不揮発性メモリ１６Ｄには、ＳＨＡ－１のハッシュアルゴリズムに従って、ハッシュ値を計算するプログラムが記憶されている。

【００１８】

通信部１６Ｆは、アンテナや無線送受信部を備え、移動パケット通信網１５と無線パケット通信を行うものであり、ＣＰＵ１６Ｂと移動パケット通信網１５との間でパケットを中継する。また、通信部１６Ｆは、通話のためのマイク、スピーカ等を備えており、これによって携帯電話機１６は図示せぬ移動電話網を介して回線交換による通話を行うこともできる。

操作部１６Ｇは操作子を備え、操作子の操作に応じた信号をＣＰＵ１６Ｂへ供給する。

【００１９】

（２．動作）

次に、上述した通信システムの動作例について説明する。

携帯電話機１６の図示せぬ電源が投入されると、ＣＰＵ１６ＢはＲＡＭ１６Ｅをワークエリアとし、ＲＯＭ１６ＡからＯＳソフトウェアに内包されているプログラムを読み出して実行する。これにより、ＣＰＵ１６ＢにはＵＩ（User Interface；ユーザインターフェース）等を提供する機能が実現される。すなわち、Ｃ

ＰＵ１６ＢはＯＳソフトウェアを起動して携帯電話機１６内にて図５に示すＯＳを実現する。ＯＳは操作部１６Ｇから供給される信号とＵＩの状態とに基づいてユーザの指示を特定し、この指示に応じた処理を行う。

【００２０】

例えば、ユーザの指示がＪａｖａ－ＡＰソフトウェアのダウンロードを要求するものである場合には、Ｗｅｂブラウザは、この指示を次に述べるＪＡＭ（Java Application Manager）に通知する。

また、ユーザの指示がネイティブＡＰソフトウェアであるＪＡＭソフトウェアの起動を要求するものであれば、ＯＳはＪＡＭソフトウェアを起動して携帯電話機１６内にてＪＡＭを実現する。ＪＡＭは、携帯電話機１６にインストールされているＪａｖａ－ＡＰソフトウェアの一覧をユーザに提示し、ユーザにより指定されたＪａｖａ－ＡＰソフトウェアを起動する。具体的には、ＪＡＭに対するユーザの指示がＪａｖａ－ＡＰソフトウェアの起動を要求するものであれば、Ｊａｖａ－ＡＰ環境ソフトウェアが起動されて携帯電話機１６内にＪａｖａ－ＡＰ環境が実現され、次に、指定されたＪａｖａ－ＡＰソフトウェアが起動されてＪａｖａ－ＡＰ環境内にＪａｖａ－ＡＰが実現される。Ｊａｖａ－ＡＰ環境は、携帯電話機１６のような携帯端末に適した軽量のＪａｖａ仮想マシンであるＫＶＭ（K Vartual Machine）と、Ｊａｖａ－ＡＰに対して提供されるＡＰＩ（Application Interface）とを有する。Ｊａｖａ－ＡＰに対して提供されるＡＰＩは、トラステッドＪａｖａ－ＡＰソフトウェアによって実現されるＪａｖａ－ＡＰに使用が許可されるＳＤＦのパーミッション情報データで指定されるＡＰＩと、あらゆるＪａｖａ－ＡＰに使用が許可される非トラステッドＡＰＩとに分けられる。

【００２１】

なお、以下に述べる動作において、ＴＣＰコネクションの確立および切断動作についてはＨＴＴＰにおける一般的な動作となることから、それらの説明を省略する。また、前述のＯＳ、Ｗｅｂブラウザ、ＪＡＭ、Ｊａｖａ－ＡＰ、ネイティブＡＰ等が行う動作は携帯電話機１６の動作となることから、以降の説明では、動作の主体を携帯電話機１６とする。

【００２２】

（２－１：トラステッドＪａｖａ－ＡＰソフトウェアの作成）

まず、ＣＰの管理するＣＰサーバ装置１２が、トラステッドＪａｖａ－ＡＰソフトウェアを作成する動作を、図６を参照して説明する。

ここで、ＣＰは、予め、入力されたＪＡＲファイルや証明書データより、ＳＨＡ－１のハッシュアルゴリズムに従ってハッシュ値を算出するプログラム（以下、「ツール」という）を通信事業者より貸与されており、当該プログラムは、ＣＰサーバ装置１２のハードディスク装置１２Ａに記憶されているものとする。また、ＣＰサーバ装置１２のハードディスク装置１２Ａには、予めＣＡより取得した証明書データが記憶されているものとする。

【００２３】

まず、ＣＰは、ＣＰが所望するアプリケーション「TELNO別着信メロディ変更アプリ」を実現させるためのプログラムを作成し、当該プログラムを内包したＪＡＲファイルを“http://www.b.co.jp/ melody.jar”で特定される位置に記憶させるよう入力する。また、ＣＰは、当該プログラムのアプリ名“melody　by TELNO”を表すアプリ名データやＪＡＲ保存先ＵＲＬデータ等の各種情報を記述したＡＤＦをＣＰサーバ装置１２に入力し、通信事業者より貸与されているツールを起動する指示を行う。

これにより、ＣＰサーバ装置１２のＣＰＵは、ハードディスク装置１２Ａよりツールを読み出し、入力されたＪＡＲファイルより、Ｊａｒハッシュ値データを算出する。また、ＣＰサーバ装置１２のＣＰＵは、証明書データより、証明書データハッシュ値を算出する。そして、ＣＰサーバ装置１２のＣＰＵは、算出したＪａｒハッシュ値データと、証明書データ、アプリ名データ“melody　by TELNO”、ＪＡＲ保存先ＵＲＬデータ“http://www.b.co.jp/ melody.jar”とを含んだＡＤＦを作成する。

そして、ＣＰサーバ装置１２は、トラステッドサーバ装置１８に、作成したＡＤＦ、ＪＡＲファイル、証明書データハッシュ値とを送信する（ステップＳ１）。

【００２４】

トラステッドサーバ装置１８を所有する通信事業者は、上記ファイルを受信した後、まず、当該ファイルを作成したＣＰの安全性を審査する。

具体的には、通信事業者は、ＡＤＦに内包されている証明書データは、通信事業者が認めた正当なＣＡから発行されているか、証明書が複数のＣＡから発行されている場合には、当該複数の証明書によって構成される証明書チェーンは正当であるか、チェーンの上位の認証機関は通信業者が認めたＣＡであるか等の検証を行う。

次に、通信事業者は、ＣＰが作成したＪａｖａ－ＡＰソフトウェアの安全性を審査する。具体的には、例えば、通信事業者は、ＪＡＲファイルのプログラム記述を検査することにより、当該Ｊａｖａ－ＡＰソフトウェアにより実現されるアプリケーションが、携帯電話機１６に記憶されている個人情報を破壊したり、個人情報を流出させたりする可能性があるか否かを審査する。

そして、通信事業者は、上記検証結果、検査結果を基に、Ｊａｖａ－ＡＰソフトウェアが使用可能なＡＰＩやＵＲＬを表すパーミッション情報を決定する。通信事業者は、これらのデータをトラステッドサーバ装置１８に入力する。

【００２５】

トラステッドサーバ装置１８のＣＰＵは、ＣＰサーバ装置１２より受信したＡＤＦ、ＪＡＲファイルに対応するＳＤＦを生成する。

具体的には、ＣＰＵは、ＡＤＦファイルから読み出した“http://www.b.co.jp/melody.jar”を表すＪＡＲ保存先ＵＲＬデータと、“ＹＥＳ”を表すＡＤＦ証明書フラグデータと、ＣＰサーバ装置１２より受信した証明書ハッシュ値データと、通信事業者によって入力されたパーミッション情報データとを含んだＳＤＦを生成する。

【００２６】

次に、トラステッドサーバ装置１８のＣＰＵは、受信したＡＤＦに、トラステッドＪａｖａ－ＡＰソフトウェアを識別するためのトラステッドＡＰＩＤデータ“０００１”と、対応するＳＤＦが記憶されているトラステッドサーバ装置１８における位置を特定するためのトラステッドサーバドメインデータ“http://www.a.co.jp/melody.sdf”を付加する。そして、ＣＰＵは、当該データを付加した

ＡＤＦをＣＰサーバ装置１２に送信する（ステップＳ２）。

【００２７】

ＣＰサーバ装置１２のＣＰＵは、ＡＤＦを受信して、自装置のハードディスク装置１２Ａに当該ＡＤＦを記憶する。これにより、Ｊａｖａ－ＡＰソフトウェアは携帯電話機１６よりダウンロード可能な状態となる。

【００２８】

（２－２：携帯電話機１６によるＪａｖａ－ＡＰソフトウェアのダウンロード）

次に、ユーザが携帯電話機１６を用いて、Ｊａｖａ－ＡＰソフトウェアのダウンロードの指示を行ったときの動作を、図６を参照して説明する。

ここでは、上記２－１：トラステッドＪａｖａ－ＡＰソフトウェアの作成処理が行われてから、ＡＤＦ、ＪＡＲファイルの内容は、変更されていないものとする。

【００２９】

ユーザは、携帯電話機１６の操作部１６Ｇを操作して、ＣＰサーバ装置１２よりトラステッドＪａｖａ－ＡＰソフトウェア「TELNO別着信メロディ変更アプリ」をダウンロードする指示を行う。

ＣＰＵ１６Ｂは、上記トラステッドＪａｖａ－ＡＰソフトウェアのダウンロードを要求する指示がＷｅｂブラウザから通知されると、当該トラステッドＪａｖａ－ＡＰソフトウェアを携帯電話機１６にダウンロードする処理を行う。

まず、ＣＰＵ１６Ｂは、ダウンロードしようとするＪａｖａ－ＡＰソフトウェアに対応するＡＤＦをＣＰサーバ装置１２から取得する。具体的には、ＣＰＵ１６Ｂは、ＣＰサーバ装置１２との間にＴＣＰコネクションを確立し、このＡＤＦの送信を要求する内容の要求メッセージを生成・送信し（ステップＳ３）、このメッセージに対する応答メッセージを受信してＡＤＦを取得した後（ステップＳ４）、このＴＣＰコネクションを切断する。そして、ＣＰＵ１６Ｂは、応答メッセージに内包されているＡＤＦを不揮発性メモリ１６Ｄに書き込む。

【００３０】

次いで、ＣＰＵ１６Ｂは、ダウンロードしようとするＪａｖａ－ＡＰソフトウェアがトラステッドＪａｖａ－ＡＰソフトウェアであるか否かを判定する。具体

的には、ＣＰＵ１６Ｂは、受信したＡＤＦ内にトラステッドＡＰＩＤデータが記述されているか否かを確認し、記述されていれば、このＪａｖａ－ＡＰソフトウェアに対応するＳＤＦが存在する、即ち、トラステッドＪａｖａ－ＡＰソフトウェアであると判定し、その記述がなければ非トラステッドＪａｖａ－ＡＰソフトウェアであると判定する。

【００３１】

そして、ダウンロードしようとするＪａｖａ－ＡＰソフトウェアが非トラステッドＪａｖａ－ＡＰソフトウェアであると判定された場合には、ＡＤＦに内包されているＪＡＲ保存先ＵＲＬデータで特定されるＵＲＬで表される位置より、ＪＡＲファイルがダウンロードされ、従来と同様のダウンロード処理が行われる。

【００３２】

ここでは、トラステッドＡＰＩＤデータに“０００１”を表すデータが記述されているので、ＣＰＵ１６Ｂは、ダウンロードしようとするＪａｖａ－ＡＰソフトウェアがトラステッドＪａｖａ－ＡＰソフトウェアであると判定し、ＡＤＦに内包されているトラステッドサーバドメインデータで表されるＵＲＬ“http://www.a.co.jp/ melody.sdf”で特定される位置より、このソフトウェアに対応するＳＤＦを取得する。すなわち、ＣＰＵ１６Ｂは、トラステッドサーバ装置１８との間にＴＣＰコネクションを確立し、このコネクションを介して、ＡＤＦ内に内包されているトラステッドサーバドメインデータで表されるＵＲＬ ”http://www.a.co.jp/melody.sdf”で特定されるＳＤＦの送信をトラステッドサーバ装置１８に要求する内容の要求メッセージを生成・送信する（ステップＳ５）。ＣＰＵ１６Ｂは、このメッセージに対する応答メッセージを受信してＳＤＦを取得した後（ステップＳ６）、上記コネクションを切断する。

【００３３】

次に、携帯電話機１６がＳＤＦを受信してからトラステッドＪａｖａ－ＡＰソフトウェアを検証する処理の動作を、図７を参照して説明する。

まず、ＣＰＵ１６Ｂは、ＡＤＦに証明書データが内包されているか否かを判定する（ステップＳ１０１）。具体的には、受信したＳＤＦに内包されているＡＤＦ証明書有無フラグデータが、“Ｙｅｓ”を表すデータか否かを判定する。ＡＤ

Ｆに証明書データが内包されていないと判定された場合には（ステップＳ１０１；Ｎｏ）、証明書データの検証は行わずに、ＪＡＲファイルのダウンロードを行う（ステップＳ１０４）。

【００３４】

ここでは、ＡＤＦ証明書有無フラグデータが“Ｙｅｓ”を表すデータであるため、ＣＰＵ１６Ｂは、ＡＤＦに証明書データが内包されていると判定し（ステップＳ１０１；Ｙｅｓ）、ＡＤＦに内包されている証明書データのハッシュ値を算出する（ステップＳ１０２）。

そして、ＣＰＵ１６Ｂは、ＡＤＦに内包されている証明書データから算出したハッシュ値と、ＳＤＦに予め内包されている証明書ハッシュ値データで表される証明書ハッシュ値とを比較し、一致しているか否か判定する（ステップＳ１０３）。一致していない場合には（ステップＳ１０３；Ｎｏ）、通信事業者がＳＤＦを作成した時点以降に、ＡＤＦに内包されている証明書データが変更されている可能性があるため、ユーザにダウンロード失敗の通知を行うと共に、ＣＰＵ１６ＢはＡＤＦを削除して、携帯電話機１６をＡＤＦダウンロードする前の状態に戻し（ステップＳ１０７）、処理を終了する。

【００３５】

ここでは、ハッシュ値は一致しているため（ステップＳ１０３；Ｙｅｓ）、ＣＰＵ１６Ｂは、ＪＡＲファイルをダウンロードする（ステップＳ１０４）。具体的には、ＣＰＵ１６Ｂは、ＡＤＦに内包されているＪＡＲ保存先ＵＲＬデータで表されるＵＲＬ“http://www.b.co.jp/ melody .jar”で特定されるＪＡＲファイルを記憶したＣＰサーバ装置１２との間にＴＣＰコネクションを確立し、このＪＡＲファイルの送信を要求する内容の要求メッセージを生成・送信し（図６のステップＳ７）、このメッセージに対する応答メッセージを受信してＪＡＲファイルを取得し（図６のステップＳ８）、このＴＣＰコネクションを切断する。

【００３６】

次に、ＣＰＵ１６Ｂは、ダウンロードしたＪＡＲファイルのハッシュ値を算出する（ステップＳ１０５）。そして、ＣＰＵ１６Ｂは、算出したハッシュ値と、ＡＤＦファイルに内包されているＪａｒハッシュ値データで表されるハッシュ値

とを比較し、ハッシュ値が一致しているか否かを判定する（ステップＳ１０６）。ハッシュ値が一致していない場合には（ステップＳ１０６；Ｎｏ）、ＪＡＲファイルが作成された時点以降に、ＪＡＲファイルが改竄、変更等されている可能性があるため、ＣＰＵ１６Ｂは、ダウンロードに失敗した旨をユーザに通知するとともに、ダウンロードしたＪＡＲファイルとＡＤＦとを削除して携帯電話機１６の状態をＡＤＦダウンロード以前の状態に戻し（ステップＳ１０８）、処理を終了する。

【００３７】

ここでは、ハッシュ値が一致しているため（ステップＳ１０６；Ｙｅｓ）、ＣＰＵ１６Ｂは、Ｊａｖａ－ＡＰソフトウェア取得に成功した旨をユーザに通知すると共に、取得したＪＡＲファイル、ＳＤＦを不揮発性メモリ１６Ｄに書き込み（ステップＳ１０９）、処理を終了する。

以降、ＣＰＵ１６は、トラステッドＪａｖａ－ＡＰソフトウェア「TELNO別着信メロディ変更アプリ」を実行するに際し、ＪＡＭによって、トラステッドＪａｖａ－ＡＰの挙動を監視し、パーミッション情報データに含まれる個人情報取得データ、設定更新データ、アクセス許可ＵＲＬデータによって、電話帳参照ＡＰＩ、移動機設定更新用ＡＰＩ、ＵＲＬ等の使用を許可／制限する。

【００３８】

以上説明したように、本実施形態によれば、ＡＤＦに内包されている証明書データから算出したハッシュ値と、予めＳＤＦに算出して記憶しておいたＡＤＦに内包されていた証明書データのハッシュ値とが一致しているのを確認することによって、通信事業者がＣＰから申請された証明書データを検証／許可した後にＡＤＦ内の証明書データが変更、改竄されていないことを確認することができる。つまり、ＳＤＦとＡＤＦとの組み合わせの正当性を判定することができる。なお、ＡＤＦ内の証明書データが同一の証明書データで上書きされた場合にはハッシュ値が一致することになり、ＡＤＦがＳＤＦ作成時のＡＤＦと異なる場合にもＳＤＦとＡＤＦとの組み合わせが正当であると判定されることになる。つまり、上記の検証／許可の後にＣＰがＡＤＦを変更しても、ＳＤＦとＡＤＦとの組み合わせは正当と判定される。換言すれば、ＳＤＦとＡＤＦとの組み合わせについては

広い意味での正当性が判定される。これにより、ＣＰは、バグフィックス等のためにＣＰがＪＡＲファイル及びＡＤＦを変更しても、上記の検証／許可をやり直さなくてよい。

また、本実施形態によれば、ＪＡＲファイルから算出したハッシュ値と、予めＡＤＦに算出して記憶しておいたＪＡＲファイルのハッシュ値とが一致していることを確認することによって、ＡＤＦとＪＡＲファイルとの組み合わせの正当性を判定することができる。これにより、例えば、第３者によって改竄されたＪＡＲファイルの起動を禁止したりすることができる。

また、本実施形態によれば、記録媒体を予め配布したりせずとも、移動機において、ファイルの組み合わせの正当性を判定することができる。

このように、本実施形態によれば、ダウンロードした関連するＡＤＦ、ＳＤＦ及びＪＡＲファイルの組み合わせの正当性を容易に判定することができる。

また、ハッシュ値の算出処理は公開鍵を用いた一般的な認証処理に比較して遥かに軽いから、本実施形態によれば、移動機のように情報処理能力が低い装置であっても、組み合わせの正当性の判定を容易に行うことができる。

【００３９】

（３：変形例）

本発明は上述した実施形態に限定されず、以下のような種々の変更が可能である。

【００４０】

（１）上記実施形態においては、ＡＤＦに内包されている証明書データのハッシュ値をこのＡＤＦに対応するＳＤＦに内包させる一方、携帯電話機１６においてＡＤＦの証明書データのハッシュ値を算出し、これら両者を比較してＳＤＦとＡＤＦの対応関係の正当性を確認するようにしていたが、ＡＤＦ内の証明書データ以外のデータのハッシュ値を用いるように実施形態を変形してもよい。例えば、ＡＤＦ全体のハッシュ値を用いるようにしてもよい。

また、同様に、上記の実施形態においては、ＪＡＲファイルのハッシュ値をこのＪＡＲファイルに対応するＡＤＦに内包させる一方、携帯電話機においてＪＡＲファイルのハッシュ値を算出し、これら両者を比較してＪＡＲファイルとＡＤ

Ｆとの対応関係の正当性を確認するようにしていたが、例えば、ＪＡＲファイルの一部のハッシュ値を用いるように実施形態を変形してもよい。

【００４１】

（２）上記実施形態においては、SHA-1のアルゴリズムによるハッシュ関数を用いてハッシュ値を算出することにより、ダウンロードした関連する複数のファイルの正当性を判定したが、用いるハッシュ関数は、これに限定されない。例えば、MD5のアルゴリズムによるハッシュ関数を用いてもよい。また、ハッシュ関数に限らず、任意の一方向関数を用いることができる。

【００４２】

（３）上記実施形態においては、移動機において判定を行うプログラムをＲＯＭに記憶させておくようにしたが、ＥＥＰＲＯＭに記憶させておくようにしてもよいし、移動機通信網を介してダウンロードしてＥＥＰＲＯＭに書き込むようにしてもよい。また、当該プログラムを記録した記録媒体を移動機に装着し、このプログラムを移動機が実行するようにしてもよい。

【００４３】

（４）上記実施形態においては、トラステッドサーバドメインデータをＡＤＦ内に内包させて、トラステッドＪａｖａ－ＡＰソフトウェアに対応するＳＤＦを特定するようにしたが、ＳＤＦを特定する方法はこれに限定されない。例えば、トラステッドサーバ装置１８を特定するためのＵＲＬを表すデータを予め携帯電話機１６に記憶させておき、トラステッドサーバ装置１８にＳＤＦの送信を要求するための要求メッセージを作成する際に、トラステッドサーバ装置１８を特定するためのＵＲＬを表すデータと、当該ＵＲＬにおいてトラステッドＪａｖａ－ＡＰソフトウェアに対応するＳＤＦ（のファイル名）を識別するためのトラステッドＡＰＩＤデータとを内包させることによって、ＳＤＦを特定するようにしてもよい。

【００４４】

【発明の効果】

本発明によれば、ダウンロードした関連する複数のファイルの組み合わせの正当性を判定することができる。

【図面の簡単な説明】

【図１】　本発明の実施の一形態に係る配信システムの構成を示すブロック図である。

【図２】　同システムに特有のＡＤＦのデータ構成を示す概念図である。

【図３】　同システムにおいてトラステッドサーバ装置に記憶されているＳＤＦのデータ構成を示す概念図である。

【図４】　同システムを構成する携帯電話機の構成を示すブロック図である。

【図５】　同携帯電話機の機能構成を示す概念図である。

【図６】　本発明の実施形態におけるデータの流れを示すためのシーケンスーチャートである。

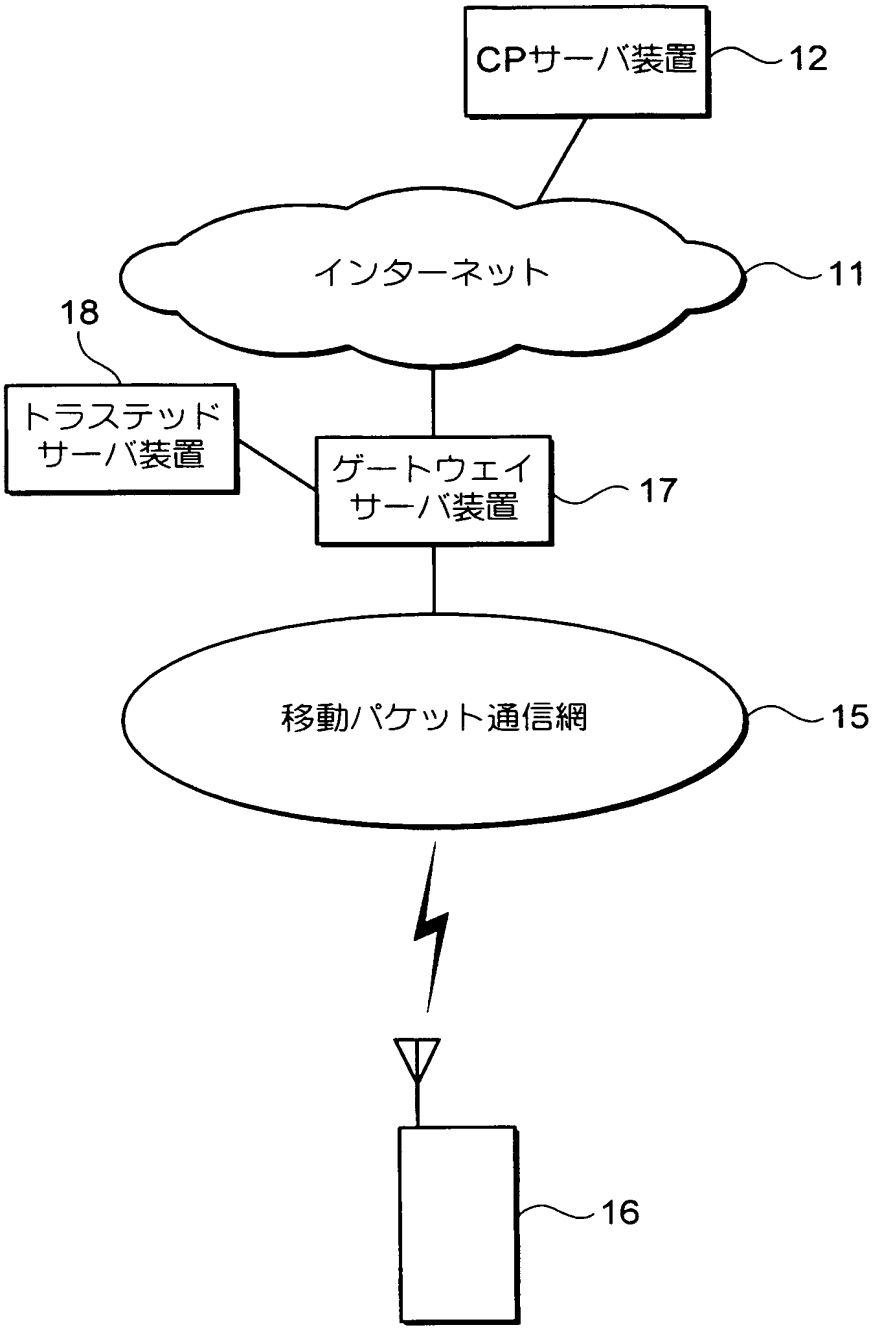
【図７】　同携帯電話機の検証処理の流れを示すためのフローチャートである。

【符号の説明】

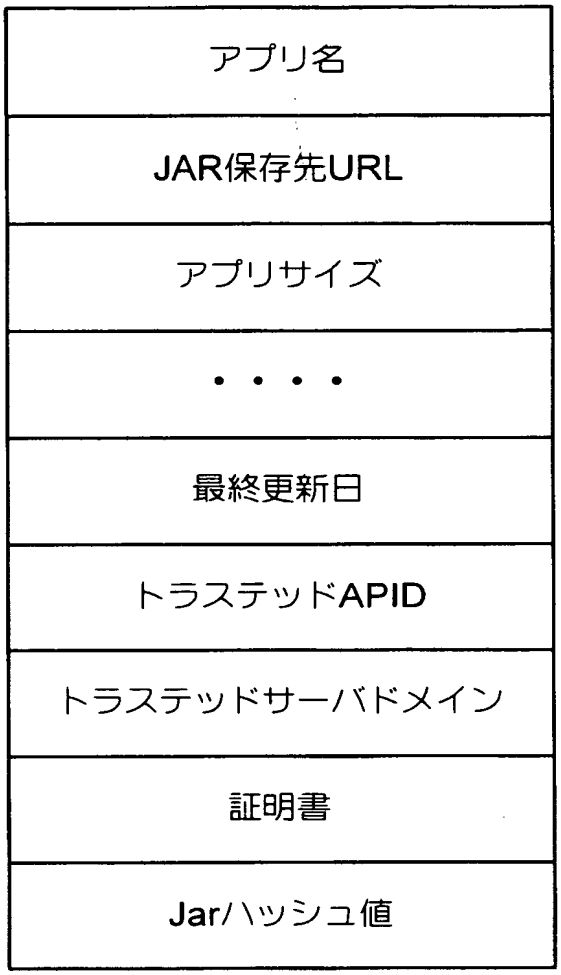
１１　インターネット、１２　ＣＰサーバ装置、１５　移動パケット通信網、１６　携帯電話機、１７　ゲートウェイサーバ装置、１８　トラステッドサーバ装置、１６Ｄ　不揮発性メモリ、１６Ａ　ＲＯＭ、１６Ｂ　ＣＰＵ、１６Ｃ　表示部、１６Ｅ　ＲＡＭ、１６Ｆ　通信部、１６Ｇ　操作部。

【書類名】　図面

【図１】


CPサーバ装置
12
インターネット
11
18
トラステッド
サーバ装置
ゲートウェイ
サーバ装置
17
移動パケット通信網
15
16

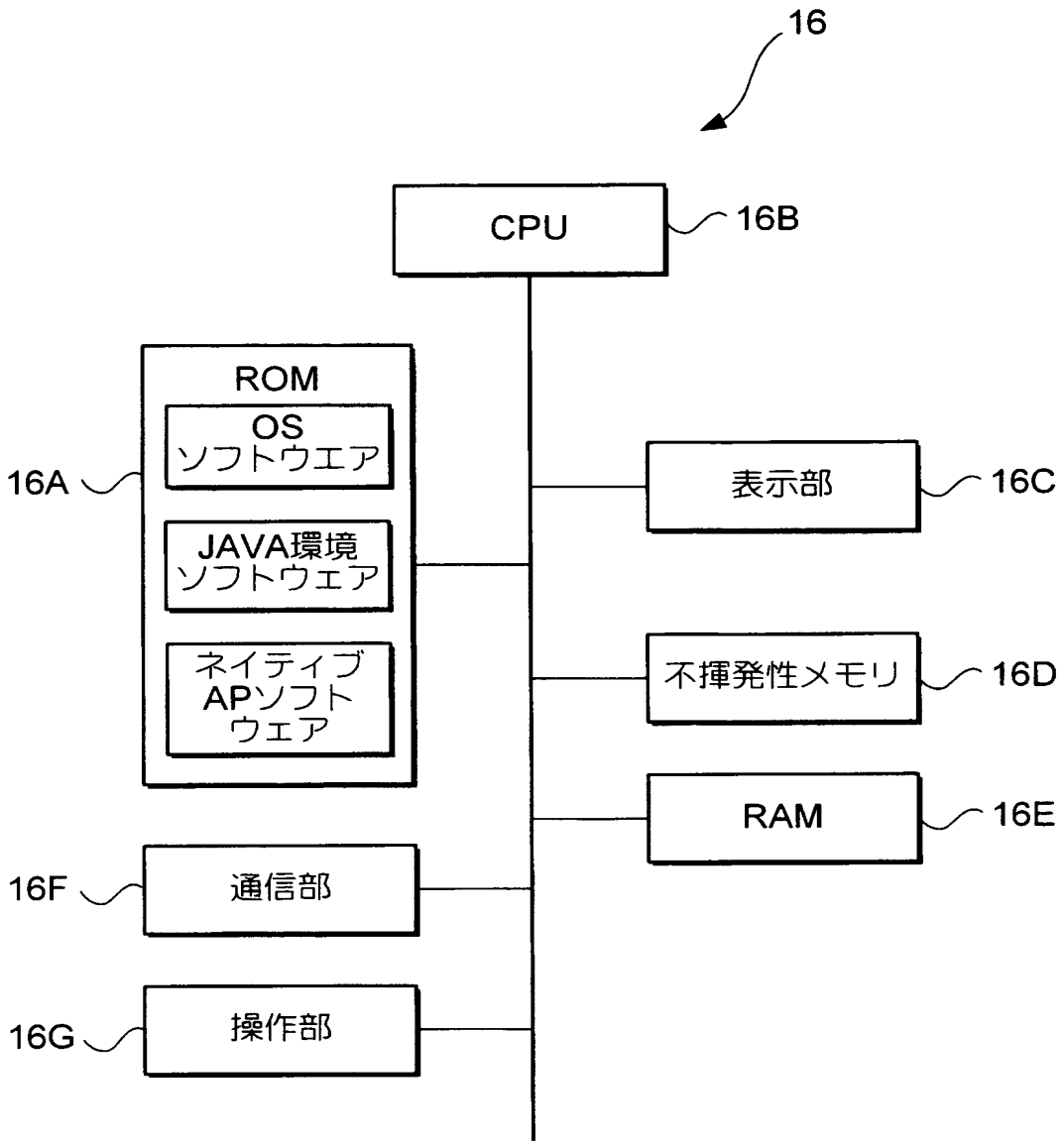
【図２】


アプリ名
JAR保存先URL
アプリサイズ
・・・・
最終更新日
トラステッドAPID
トラステッドサーバドメイン
証明書
Jarハッシュ値

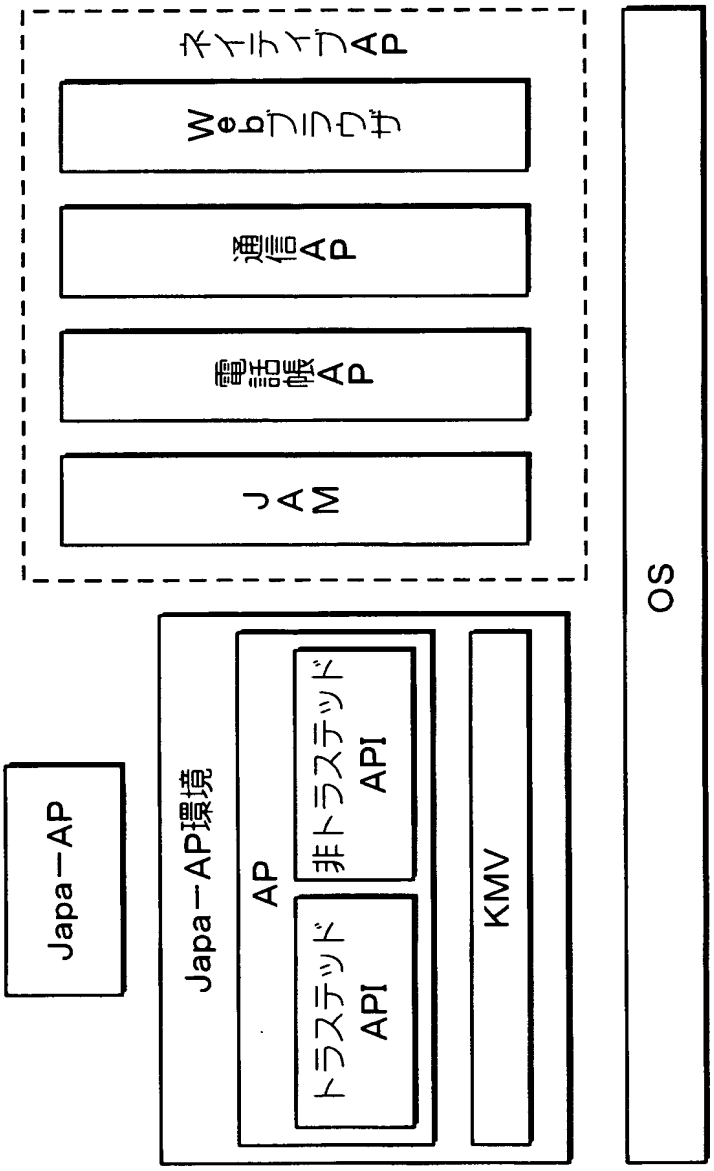
【図３】


JAR保存先URL
ADF証明書有無フラグ
証明書ハッシュ値
パーミッション情報
個人情報取得
設定更新
アクセス許可URL

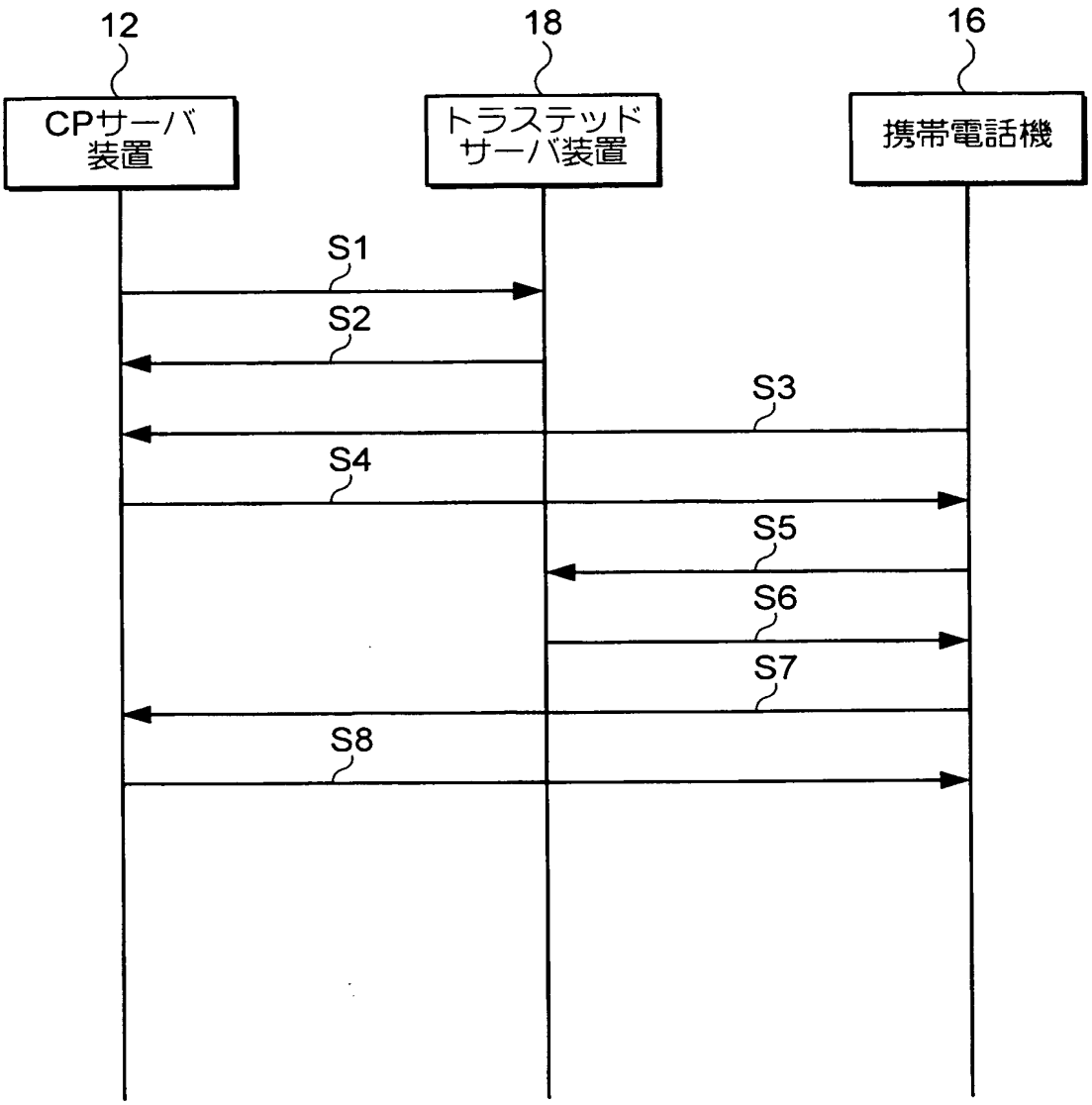
【図４】


16
CPU
16B
ROM
OS
ソフトウエア
16A
JAVA環境
ソフトウエア
ネイティブ
APソフト
ウエア
表示部
16C
不揮発性メモリ
16D
RAM
16E
16F
通信部
16G
操作部

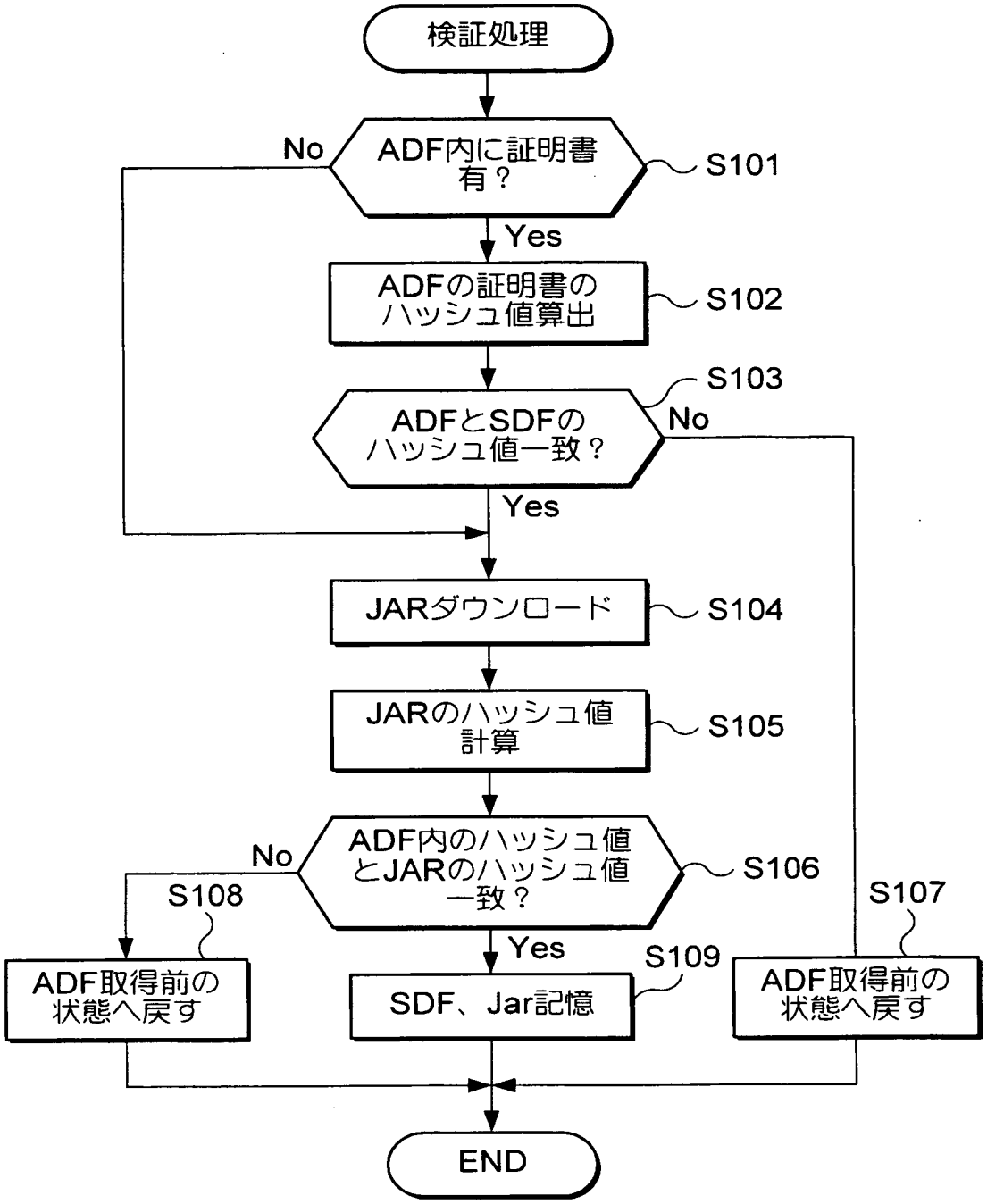
【図５】


ネットワークAP
Webブラウザ
通信AP
電話帳AP
JAM
OS
Japa－AP環境
AP
非トラステッド API
トラステッド API
KMV
Japa－AP

【図６】



【図７】


検証処理
ADF内に証明書有？
No
S101
Yes
ADFの証明書のハッシュ値算出
S102
S103
ADFとSDFのハッシュ値一致？
No
Yes
JARダウンロード
S104
JARのハッシュ値計算
S105
ADF内のハッシュ値とJARのハッシュ値一致？
No
S106
S108
S107
Yes
S109
ADF取得前の状態へ戻す
SDF、Jar記憶
ADF取得前の状態へ戻す
END

【書類名】　　要約書

【要約】

【課題】　本発明は、ダウンロードした関連する複数のファイルの組み合わせの正当性を容易に判定することができる技術を提供する。

【解決手段】　Ｊａｖａ－ＡＰソフトウェアを起動することができる携帯電話機１６が、ＣＰサーバ装置１２からＡＤＦを取得し、このＡＤＦを用いてトラステッドサーバ装置１８からＳＤＦを受信し、ＡＤＦに内包されている証明書データから算出したハッシュ値とＳＤＦに内包されている予め算出しておいたハッシュ値とが一致していることを確認する。次いで、携帯電話機１６は、ＣＰサーバ装置１２からＪＡＲファイルを取得し、ＪＡＲファイルから算出したハッシュ値とＡＤＦに内包されている予め算出しておいたハッシュ値とが一致していることを確認する。

【選択図】　　図１

特願２００３－０９６０８８

出　願　人　履　歴　情　報

識別番号　［３９２０２６６９３］

１．変更年月日　２０００年　５月１９日
［変更理由］　名称変更
住所変更
住　所　東京都千代田区永田町二丁目１１番１号
氏　名　株式会社エヌ・ティ・ティ・ドコモ